

It-arkitektur – Agil udvikling

Opgave 1: Mocks og stubs (igen)

Hvorfor bruger vi mocks og stubs.

Fremhæv fordele og ulemper.

Opgave 2: "Flaky" tests (igen)

Er det i orden, at fjerne tests "bare" fordi de er "flaky"?

Opgave 3: NUnit tutorial (igen)

Gennemfør <https://learn.microsoft.com/en-us/dotnet/core/testing/unit-testing-with-nunit>

Opgave 4: Billetpriser – TDD / Test first (igen)

Et busselskab har nogle underlige regler for prisberegning: Prisen afhænger af den kørte afstand og antallet af passagerer. Reglerne er som følger: Der er et startgebyr på 230 kr. Alt under 100 km koster 6,20 kr. per kilometer, uanset passagerantal. For kilometerne over 100 og mindre end 500 betalte 5,40 kr. for hver kilometer, hvis der er mindre end 12 passagerer. Hvis der er mere end 12 passagerer koster det 6,00 kr per kilometer. For kilometerne over 500 koster hver kilometer 4,25 kr.

I skal implementere en funktion der indkapsler denne logik og samtidig lave de tilhørende unit-tests vha. test-drevet udvikling.

```
/* Returnerer prisen for en tur hvor der er [passengers] passagerer og  
   der køres [distance] kilometer. Begge input parametre kan antages  
   at være heltal og ikke negative. */
```

```
double BusTicketPrice(int passengers, int distance) {...}
```

Husk, at vi følger princippet om, at vi kun skal lave den enkleste ting, der fungerer.

Har du konstateret mulige fælder i processen?

Opgave 5: Test af microservicer vedrørende taxabooking

Prøv at udarbejde test af microservicer, der kan anvendes i forbindelse med taxabooking. Du er velkommen til at inddrage andre microservicer, du har arbejdet med.

Hvad virker for dig/er teknologisk?

Hvad virker ikke for dig/er teknologisk?

Send en status til kell@eaaa.dk.

Opgave 6: Registrering af Bud i en auktion (NY!!!)

Med forbehold for jeres målbillede, så kunne man forestille sig, at I skulle implementere "Bud i en auktion".

- Hvad er kravene?
- Og hvad kan vi i en test-first strategi opstille som tests.

Opgave 7: Registrering af Bud i en auktion (NY!!!)

Med forbehold for jeres målbillede, så kunne man forestille sig, at I skulle implementere "Bud i en auktion".

- Hvad er kravene?
- Og hvad kan vi i en test-first strategi opstille som tests.

EKSTRA EXERCISE: Collection of shops

Imagine you are developing a system where a shop is an important business entity. The shop has a name, address, website, and a gps location (latitude and longitude).

The system must contain more complicates search related functionality.

1. The first of these is that shop must be shown sorted by their distance to a specific location.
2. The second one is a kind of filter; it should be possible to filter the shops by a rectangle.

In order to implement this there are a need for two functions:

```
// Return: all shops in source, sorted according to their distance to location x.  
List<BEShop> SortByDistance(List<BEShop> source, GeoCoordinate x);  
  
// Return all shops from source located in the rectangle defined by x, width and height,  
// where x is the lower left corner of the rectangle. The unit of width and height is kilometer.  
List<BEShop> FilterByLocation(List<BEShop> source, GeoCoordinate x, int width, int height);
```

GeoCoordinate is from the nuget-package GeoCoordinate.NetCore (see <https://www.nuget.org/packages/GeoCoordinate.NetCore/>)

Implement the two functions using TDD.

You can use gps.txt if you like.